



**HAL**  
open science

# Non Linear Model Predictive Control Using Constraints Satisfaction

Fabien Lydoire, Philippe Poignet

► **To cite this version:**

Fabien Lydoire, Philippe Poignet. Non Linear Model Predictive Control Using Constraints Satisfaction. COCOS: Global Optimization and Constraint Satisfaction, Nov 2003, Lausanne, Switzerland. pp.142-153, 10.1007/11425076\_11 . lirmm-00269731

**HAL Id: lirmm-00269731**

**<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00269731>**

Submitted on 20 Sep 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Nonlinear Predictive Control Using Constraints Satisfaction

Fabien Lydoire and Philippe Poignet

Laboratoire d'Informatique,  
de Robotique et de Microélectronique de Montpellier,  
UMR CNRS UM2 5506, 161 rue Ada,  
34392 Montpellier Cedex 5, France  
{lydoire, poignet}@lirmm.fr

**Keywords:** interval analysis, state estimation, nonlinear model predictive control.

## 1 Introduction

During the last few years, control schemes using interval analysis have been investigated. Several approaches have been proposed in order to get robust control in presence of model uncertainties [7, 10] or for state estimation [6].

In this paper, we investigate the design of a nonlinear model predictive controller [1], using set computation. The motivation for using NMPC control is its ability to handle nonlinear multi-variable systems that are constrained in the state and/or in the control variables. The NMPC problem is usually formulated as a nonlinear constrained optimisation one, and is solved using classic non linear optimisation techniques. However, most of the NMPC constraints are easily expressed using intervals. Therefore, we will use interval analysis techniques [8] in order to compute an NMPC constraints satisfying solution. Classic interval branch and bound algorithms have been investigated for predictive control in [3]. They conclude that the pessimism introduced by interval computation in the estimation of the states leads to high computational cost and may only be used on control of low dynamic systems. Therefore, we propose a new approach based on a spatial discretisation of the input and state domains to improve interval model predictive control and to be applied on high dynamic systems. The proposed strategy will be numerically simulated on an inverted pendulum model.

The paper is organised as follows : section 2 presents the classical nonlinear model predictive control technique, section 3 introduces interval analysis, set inversion and the proposed algorithm for its application to the NMPC problem. Finally section 4 exhibits numerical simulation results.

## 2 Nonlinear Model Predictive Control

The NMPC problem [1] is usually formulated as a constrained optimization problem

$$\min_{\mathbf{u}_k^{N_p}} J(x_k, \mathbf{u}_k^{N_p}) \quad (1)$$

subject to

$$x_{i+1|k} = f(x_{i|k}, u_{i|k}) \quad x_{0|k} = x_k \quad (2)$$

$$u_{i|k} \in \mathbb{U}, \quad i \in [0, N_p - 1] \quad (3)$$

$$x_{i|k} \in \mathbb{X}, \quad i \in [0, N_p] \quad (4)$$

where

$$\begin{aligned} \mathbb{U} &:= \{u_k \in \mathbb{R}^m \mid u_{\min} \leq u_k \leq u_{\max}\} \\ \mathbb{X} &:= \{x_k \in \mathbb{R}^m \mid x_{\min} \leq x_k \leq x_{\max}\} \end{aligned} \quad (5)$$

Internal controller variables predicted from time instance  $k$  are denoted by a double index separated by a vertical line where the second argument denotes the time instance from which the prediction is computed.  $x_k = x_{0|k}$  is the initial state of the system to be controlled at time instance  $k$  and  $\mathbf{u}_k^{N_p} = [u_{0|k}, u_{1|k}, \dots, u_{N_p-1|k}]$  an input vector.

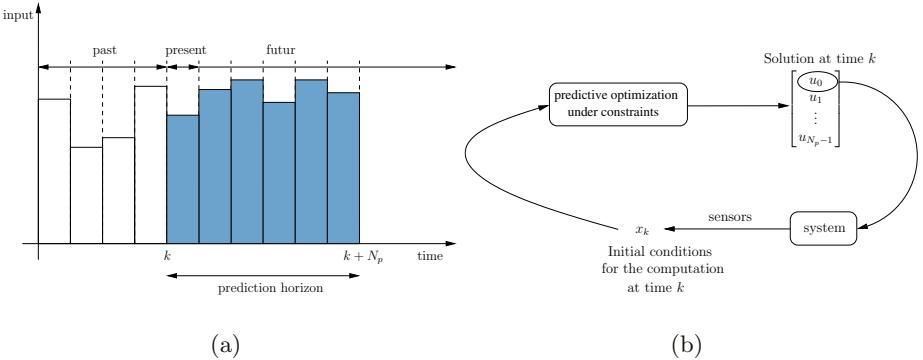
Predictive control (fig. 1) consists on computing the vector  $\mathbf{u}_k^{N_p}$  of consecutive inputs  $u_{i|k}$  over the prediction horizon  $N_p$  and applying only the solution input  $u_{0|k}$ . These computations are updated at each sampling time.

The dynamic model of the system is written as a nonlinear equality constraint on the state (eq. 2). Bounding constraints over the inputs  $u_{i|k}$  and the state variables  $x_{i|k}$  over the prediction horizon  $N_p$  are defined through the sets  $\mathbb{U}$  and  $\mathbb{X}$  (eq. 5).

The objective function  $J$  is usually defined as

$$J(x_k, \mathbf{u}_k^{N_p}) = \phi(x_{N_p|k}) + \sum_{i=0}^{N_p-1} L(x_{i|k}, u_{i|k}) \quad (6)$$

where  $\phi$  is a constraint over the state at the end of the prediction horizon, called state terminal constraint, and  $L$  a quadratic function of the state and inputs.



**Fig. 1.** Principles of the predictive constrained optimal control approach

The solution  $\mathbf{u}_k^{N_p}$  of the NMPC problem has two properties. Firstly, it satisfies the constraints over the inputs ( $u_k \in \mathbb{U}$ ) and the states ( $x_k \in \mathbb{X}$ ), including the state terminal constraint. Secondly it is optimal with respect to the criteria  $J$ . In this article, we will consider the computation of a solution satisfying the constraints, without considering the optimisation.

Except for the dynamic model of the system (eq. 2) which is nonlinear, NMPC constraints (eqs. 3,4) are inequality constraints and can directly be written as intervals. Therefore, it would be interesting to use interval techniques in order to compute a solution satisfying the NMPC constraints. The following section introduces interval analysis concepts used to compute such a solution.

### 3 Constraints Satisfaction

#### 3.1 Interval Analysis and Set Inversion

Initially dedicated to finite precision arithmetic for computer [11] and after used in a context of guaranteed global optimization [4], the interval analysis is based on the idea of enclosing real numbers in intervals and real vectors in boxes.

Let  $f$  be a function from  $\mathbb{R}^n$  to  $\mathbb{R}^m$  and let  $\mathbb{Y}$  be a subset of  $\mathbb{R}^m$ . Set inversion is the characterization of

$$\mathbb{X} = \{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) \in \mathbb{Y}\} = f^{-1}(\mathbb{Y}) \quad (7)$$

Set inversion algorithms [8] are based on consecutive bisections of an initial domain  $[\mathbf{x}]$  for  $\mathbb{X}$ . They can perform inner ( $\underline{\mathbb{X}}$ ) and outer ( $\overline{\mathbb{X}}$ ) approximation of  $\mathbb{X}$  ( $\underline{\mathbb{X}} \subset \mathbb{X} \subset \overline{\mathbb{X}}$ ). The image  $f([\mathbf{x}])$  of  $[\mathbf{x}]$  is computed and compared to  $\mathbb{Y}$ . Four cases may be encountered:

1.  $f([\mathbf{x}]) \cap \mathbb{Y} = \emptyset$ , then  $[\mathbf{x}]$  is rejected as a subset of  $\mathbb{X}$  (fig. 2(b)).
2.  $f([\mathbf{x}]) \subset \mathbb{Y}$ , then  $[\mathbf{x}]$  is a subset of  $\mathbb{X}$  and therefore  $[\mathbf{x}]$  is stored into  $\underline{\mathbb{X}}$  and  $\overline{\mathbb{X}}$ .
3.  $f([\mathbf{x}]) \not\subset \mathbb{Y}$  and  $f([\mathbf{x}]) \cap \mathbb{Y} \neq \emptyset$ , then  $[\mathbf{x}]$  may contain a part of the solution set. If its width is greater than a precision threshold  $\epsilon$ , then  $[\mathbf{x}]$  is bisected and the test is recursively applied (fig. 2(b)).
4. If the test gives the same results as in case 3, and if the width of  $[\mathbf{x}]$  is lower than  $\epsilon$ , then  $[\mathbf{x}]$  is stored into  $\overline{\mathbb{X}}$ .

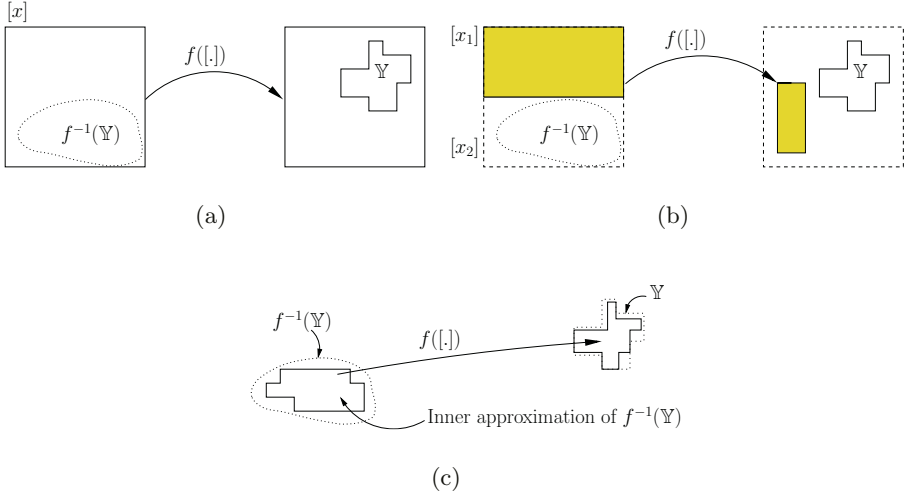
Figure 2(c) illustrates the inner approximation of  $f^{-1}(\mathbb{Y})$  finally computed by the set inversion algorithm.

Considering the initial domain  $[\mathbf{x}_{\min}, \mathbf{x}_{\max}]$ , the algorithm brackets the solution set  $\mathbb{X}' = [\mathbf{x}_{\min}, \mathbf{x}_{\max}] \cap f^{-1}(\mathbb{Y})$  by two subpavings  $\underline{\mathbb{X}}$  and  $\overline{\mathbb{X}}$ .

$$\mathbb{X}' = \{\mathbf{x} \in [\mathbf{x}_{\min}, \mathbf{x}_{\max}] \mid f(\mathbf{x}) \in \mathbb{Y}\} \subseteq f^{-1}(\mathbb{Y}) \quad (8)$$

#### 3.2 Application to the NMPC Problem

The purpose is to apply the set inversion algorithm to compute a solution satisfying the NMPC constraints.



**Fig. 2.** Set inversion algorithm steps

Considering the set inversion formulation,  $\mathbb{Y}$  domains are defined by the limits over the state variables, and the initial domain which will be bisected during the algorithm is defined by the limits over the inputs (eq. 5).

The dynamic model function  $f$  is applied over the horizon starting from the current state  $x_k$ . The computation of a new state domain  $[x_{i+1}]$  from previous state domain  $[x_i]$  and input domain  $[u_{i_{\min}}, u_{i_{\max}}]$  is followed by the set inversion algorithm (fig. 3).

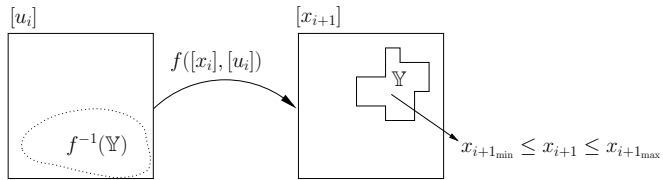
This procedure bisects the initial domain  $[u_{i_{\min}}, u_{i_{\max}}]$  and provides a domain  $[u_i]$  such that

$$f([x_i], [u_i]) = [x_{i+1}] \text{ and } [x_{i+1}] \subseteq [x_{i+1_{\min}}, x_{i+1_{\max}}] \tag{9}$$

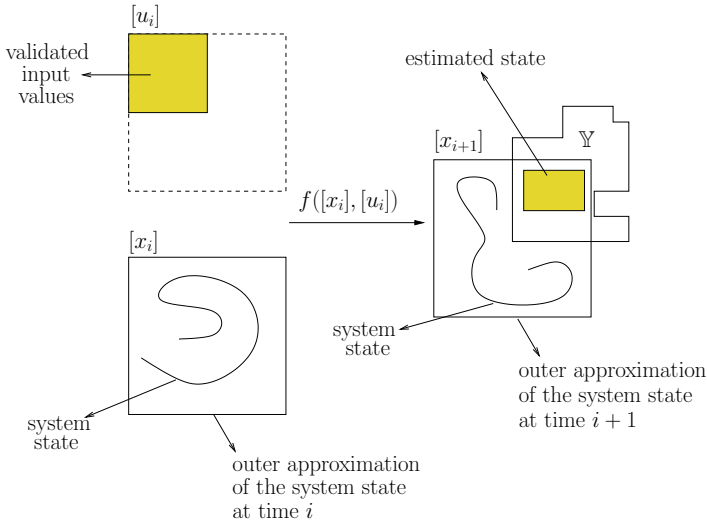
where  $[x_{i+1_{\min}}, x_{i+1_{\max}}]$  is the feasible domain for the state  $x_{i+1}$  (eq. 4). The bisection procedure reducing the width of an interval,  $[u_i]$  is such that

$$[u_i] \subseteq [u_{i_{\min}}, u_{i_{\max}}] \tag{10}$$

and therefore any punctual value in the interval  $[u_i]$  is a solution satisfying the NMPC constraints.



**Fig. 3.** Set inversion algorithm applied on NMPC



**Fig. 4.** Validation of incorrect input due to outer approximation of the state

The computation of an input satisfying the NMPC constraints implies the state estimation of the system with interval values (eq. 9). State estimation involves the computation of the dynamic model of the system followed by an integration and therefore introduces pessimism in the estimation of the states domains. State estimation on intervals are based on interval Taylor series [5, 12] and lead to guaranteed but outer approximation of the system state. Therefore the intersection of the computed state with the state constraints during the set inversion algorithm may be composed of outer state values. Consequently, the input is validated by the set inversion algorithm whereas it does not satisfy the NMPC constraints (fig. 4).

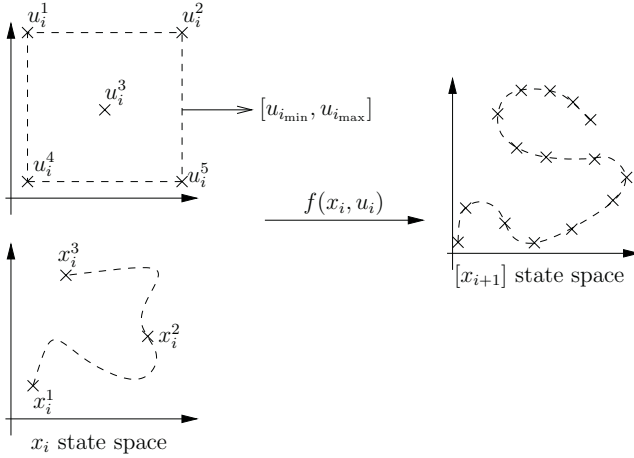
In the following, we will propose a solution to get an inner approximation of the state and thus use the set inversion algorithm to compute a NMPC constraints satisfying solution.

### 3.3 NMPC Constraints Satisfaction

Classical state estimation over intervals leads to outer approximation. However the preceding section exhibited the need for an inner approximation of the system state. Therefore, we will compute state estimation over the horizon using punctual values distributed in the considered domains.

In the following, we will omit the index  $|_k$  assuming that prediction is made at time instance  $k$ .

On each iteration, the set of inputs  $u_i^1, \dots, u_i^n$  which define a spatial distribution of the input domain  $[u_{i_{\min}}, u_{i_{\max}}]$ , is applied on each punctual state values  $x_i^1, x_i^2, \dots, x_i^m$  defining a spatial discretisation of  $[x_i]$ . This gives a new set of punctual values defining a spatial discretisation of  $[x_{i+1}]$  (fig. 5).



**Fig. 5.** Spatial discretisation

Assuming that  $f$  is continuous, the spatial discretisation of  $[x_{i+1}]$  computed by the algorithm provides an inner approximation of  $f([x_i], [u_{i_{min}}, u_{i_{max}}])$ . Indeed, for any punctual value  $x_i^p$  in  $[x_i]$ ,  $p \in [1, m]$ , and any inputs  $u_i^l$  and  $u_i^{l+1}$ ,  $l \in [1, n - 1]$  continuity of  $f$  leads to

$$[\min(f(x_i^p, u_i^l), f(x_i^p, u_i^{l+1})), \max(f(x_i^p, u_i^l), f(x_i^p, u_i^{l+1}))] \subseteq f(x_i^p, [u_i^l, u_i^{l+1}]) \quad (11)$$

therefore the set of input variables  $\mathcal{S}'$  considering the inner approximation of the state

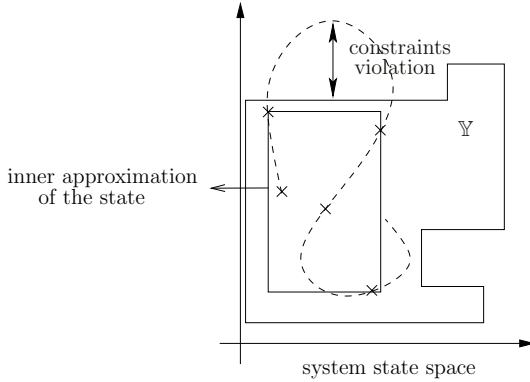
$$\begin{aligned} \mathcal{S}' = \{ & u_i \in [u_i^l, u_i^{l+1}] \mid \\ & [\min(f(x_i, u_i^l), f(x_i, u_i^{l+1})), \max(f(x_i, u_i^l), f(x_i, u_i^{l+1}))] \\ & \subseteq [x_{i+1_{min}}, x_{i+1_{max}}] \} \end{aligned} \quad (12)$$

is an inner approximation of the set of input variables  $\mathcal{S}$  in case of perfect state estimator over intervals.

$$\mathcal{S} = \{ u_i \in [u_i^l, u_i^{l+1}] \mid f(x_i, [u_i^l, u_i^{l+1}]) \subseteq [x_{i+1_{min}}, x_{i+1_{max}}] \} \quad (13)$$

The inner approximation of the state of the system allows the use of the set inversion algorithm to compute a solution satisfying the NMPC constraints. The efficiency of the solution depends on the sampled values  $u_i^l$  of the initial input interval  $[u_{i_{min}}, u_{i_{max}}]$ , and on the accuracy threshold  $\epsilon$  defining the minimum width for an interval allowed to be bisected during the set inversion procedure.

One of the drawback of the inner approximation of the state is that state values outside the inner approximation are not considered and therefore could violate the constraints (fig. 6). This leads to the validation of an incorrect input domain. However, the punctual values defining the spatial discretisation of the state are guaranteed to belong to the constrained space. These values have been



**Fig. 6.** Constraints violation due to the inner approximation of the state

computed from punctual input values defining the spatial discretisation of the input domain. Therefore, these punctual input values are guaranteed to lead to the constrained state space. However, picking any punctual value in the computed input interval may lead to constraint violation. This constraint violation has not been characterized yet and will be the object of future work.

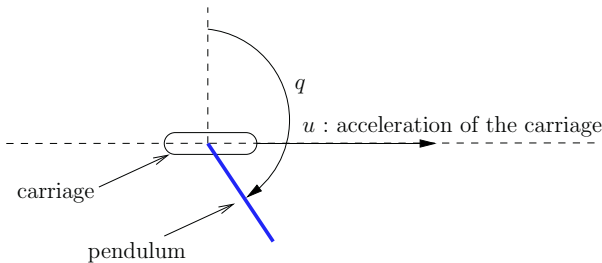
### 4 Simulation Results

The control scheme presented in this paper is applied on the stabilisation of an inverted pendulum. The pendulum is free to rotate around an horizontal axis and is actuated by a linear motor whose acceleration is the input of the system. Friction has been neglected and the hypothesis is made that the pendulum is a rigid body.

Let's consider the inverted pendulum (fig. 7) which is a classical benchmark for nonlinear control techniques [2, 9]. Its dynamic equation (eq. 2) where  $x = [q, \dot{q}]^T$  is based on the following equation

$$\ddot{q}_{t+1} = K_{\sin} \sin(q_t) - K_{\cos} u_t \cos(q_t) \tag{14}$$

Friction has been neglected and it has been assumed that the pendulum is a rigid body.



**Fig. 7.** The inverted pendulum



The acceleration  $\ddot{q}_{t+1}$  is integrated twice using:

- first order Taylor series in the predictive controller,

$$\dot{q}_{t+1} = \dot{q}_t + \delta_t \ddot{q}_{t+1} \quad (15)$$

$$q_{t+1} = q_t + \delta_t \dot{q}_{t+1} \quad (16)$$

where  $\delta_t$  is the time sampling period

- Runge-Kutta formula in the simulator.

In the simulations, a single  $[u]$  value is bisected over the horizon. Parameters  $K_{\sin}$  and  $K_{\cos}$  have been computed from a real pendulum available at the laboratory. The parameter  $\text{nb}_{\text{samples}}$  define the number of punctual values used in the spatial discretisation of  $[u]$ .  $N_p$  is the prediction horizon, the initial state is  $[q_{\text{ini}}, \dot{q}_{\text{ini}}]^T$ , the precision threshold used for bisection in the set inversion algorithm is  $\epsilon$ . The feasible values are those defined by NMPC inequalities (eqs. 3,4).

The common parameter values are regrouped in the following table

$K_{\sin}$	$K_{\cos}$	$\dot{q}_{\text{ini}}$ (rad.s <sup>-1</sup> )	$\delta_t$ (s)
109	11.11	0	0.001

$[q_{\text{feasible}}]$ (rad)	$[\dot{q}_{\text{feasible}}]$ (rad.s <sup>-1</sup> )	$[u_{\text{feasible}}]$ (m.s <sup>-2</sup> )
$[-\pi - \frac{3\pi}{2}; -\pi + \frac{3\pi}{2}]$	$[-150; 150]$	$[-800; 800]$

The punctual value  $u$  applied on the system is the closest to zero in the solution interval.

The simulations have been computed using MATLAB with a 2Ghz PENTIUM IV.

In simulations 4.1 to 4.2, the computation of the domain  $[u]$  is stopped as two valid punctual values defining the spatial discretisation of  $[u]$  have been determined. In simulations 4.4, the computation of  $[u]$  is achieved completely.

#### 4.1 Initial Position Downwards

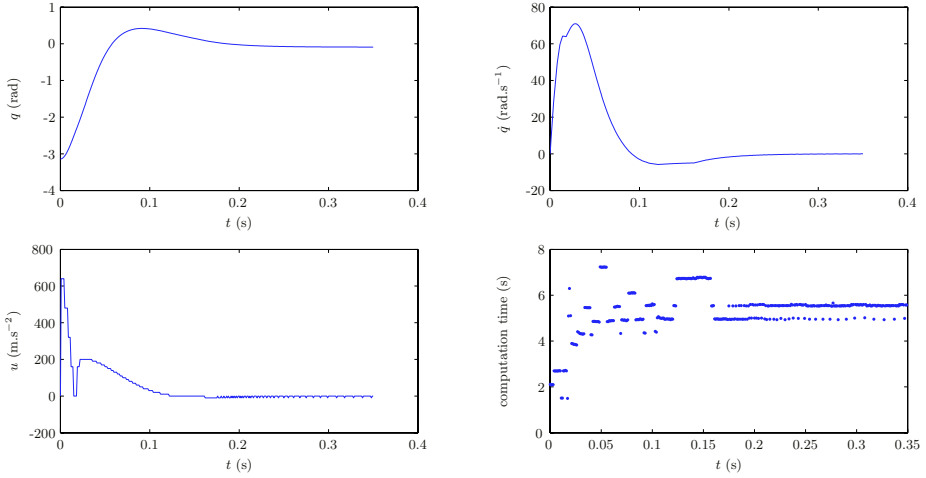
This simulation has been executed with the initial position downwards.

$N_p$	$q_{\text{ini}}$ (rad)	$\epsilon$ (m.s <sup>-2</sup> )	$[q_{N_p}]$ (rad)	$\text{nb}_{\text{samples}}$
40	$-\pi$	1.0	$[-0.1; 0.1]$	5

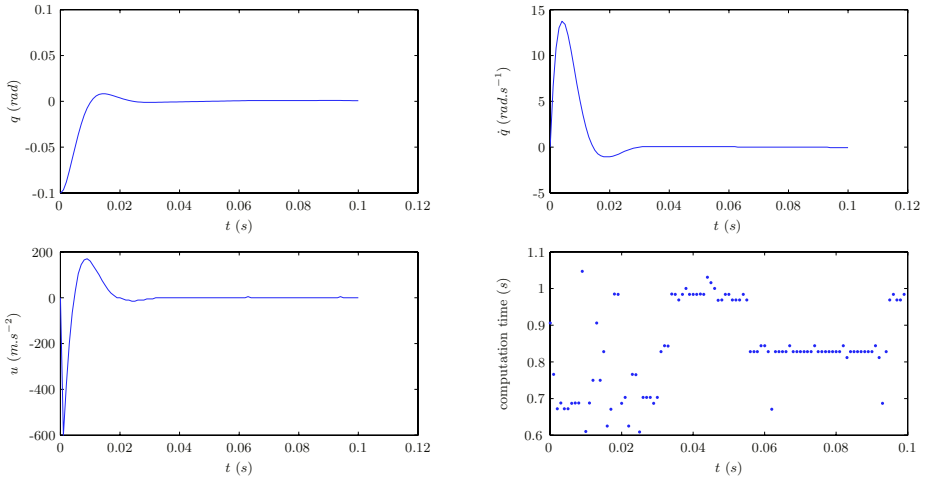
Figure 8 displays the results of this simulation. The pendulum starts from initial position  $-\pi$  and is stabilised by the control law in its terminal position  $q = [-0.1; 0.1]$  rad.

#### 4.2 Initial Position Close to 0 Rad

This simulation has been executed with the initial position close to the terminal position. It exhibits the computation time variation due to the reduction of the prediction horizon.



**Fig. 8.** Joint position, velocity, input and computation time of the input



**Fig. 9.** Joint position, velocity, input and computation time of the input

$N_p$	$q_{ini}$ (rad)	$\epsilon$ (m.s <sup>-2</sup> )	$[q_{N_p}]$ (rad)	nb <sub>samples</sub>
5	-0.1	1.0	[-0.001;0.001]	5

Figure 9 displays the results of this simulation. As in the previous simulation, the pendulum is stabilised in its terminal position. However, the computation time is reduced by a factor  $\sim 6$ .

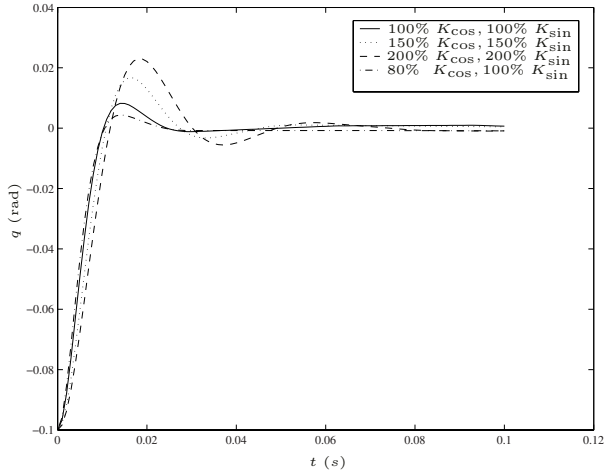


Fig. 10. Simulations with model parameters errors

### 4.3 Robustness with Respect to Model Error

The following simulations have been executed with the parameters used in simulation 4.2. Model error have been introduced through errors on the parameters  $K_{\sin}$  and  $K_{\cos}$ .

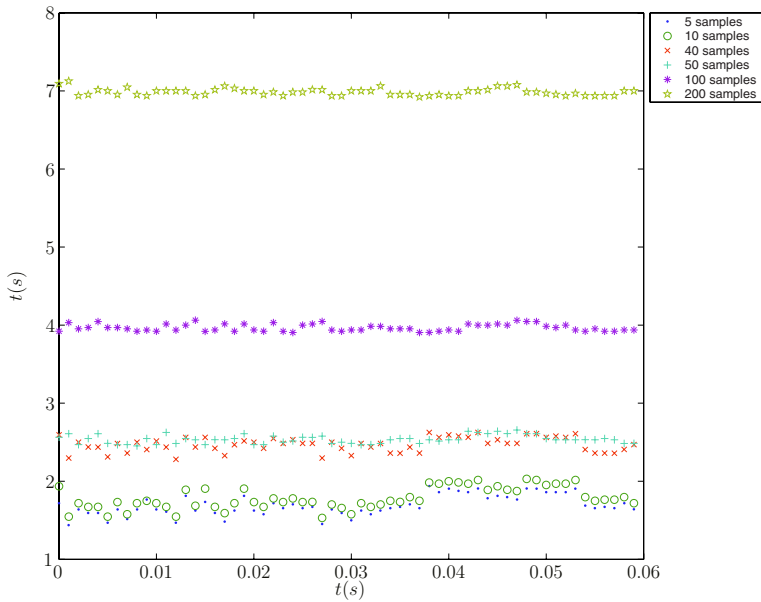
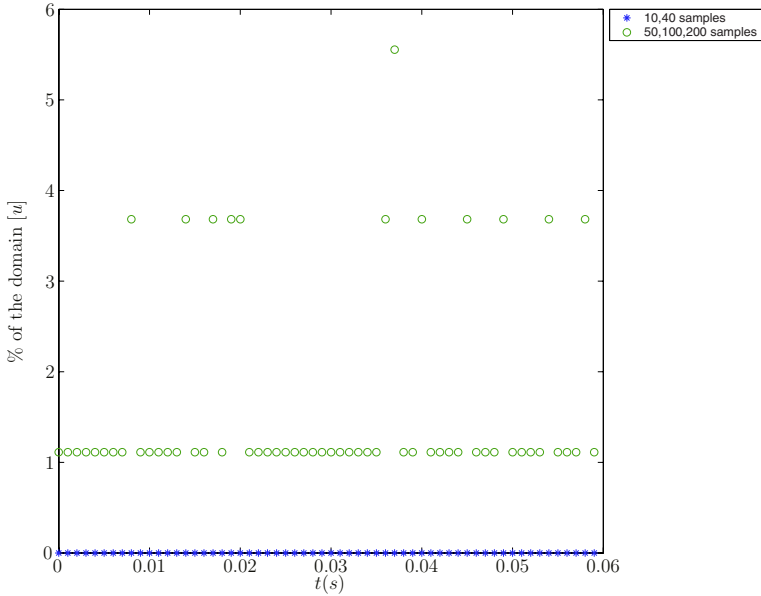


Fig. 11. Computation time with different  $nb_{\text{samples}}$  values



**Fig. 12.** Domain size percentage with respect to the domain size with  $nb_{\text{samples}} = 5$

Figure 10 exhibits the robustness of the method by displaying the joint positions. In each presented case, the control method leads the pendulum to the final constrained position. In the case of a value inferior or equal of 70% of the exact model value for  $K_{\text{cos}}$ , the algorithm is unable to find a solution.

#### 4.4 Spatial Discretisation Variation

The simulations presented in this section exhibit the influence of the parameter  $nb_{\text{samples}}$  on the calculation of the domain  $[u]$  and on computation time. The more samples there is, the longer is the computation time (fig. 11). However, the computed domain for  $[u]$  is not increased a lot (fig. 12). This is due to the algorithm used. Whatever the number of samples, the domain will be bisected until the bisected domains will be too small ( $< \epsilon$ ) to be bisected. Increasing the number of samples avoid bisections but introduces much more small domains to deal with.

### 5 Conclusion

This paper introduces a nonlinear control approach associated with interval analysis. The guaranteed state estimation techniques have been demonstrated to be inappropriate. Therefore, an inner bounding state estimation method for continuous systems has been presented. The complete simulation results show the efficiency and the robustness of the proposed method.

Future work will concern the following two points. Firstly, the computational efficiency improvement by taking into account contraction procedure based on constraints propagation. Secondly, the characterisation of the inner approximation of the state in order to compute input boxes satisfying the constraints completely.

## References

1. Frank Allgöwer, Thomas A. Badgwell, S. Joe Qin, James B. Rawlings, and Steven J. Wright. Nonlinear predictive control and moving horizon estimation - an introductory overview. In P.M. Frank, editor, *Advances in Control: Highlights of ECC '99*, chapter 12, pages 391–449. Springer-Verlag, 1999.
2. Karl Johan Åström and Katsuhisa Furuta. Swinging up a pendulum by energy control. *Automatica*, 36:287–295, 2000.
3. J.M. Bravo, C.G. Varet, and E.F. Camacho. Interval model predictive control. In *IFAC Algorithm and Architectures for Real-Time Control*, 2000.
4. Eldon R. Hansen. *Global Optimization Using Interval Analysis*. Marcel Dekker, New York, NY, 1992.
5. Kenneth R. Jackson and Nedialko S. Nedialkov. Some recent advances in validated methods for ivps for odes. *Applied Numerical Mathematics*, 42:269–284, 2002.
6. Luc Jaulin. Nonlinear bounded-error state estimation of continuous-time systems. *Automatica*, 38:1079–1082, 2002.
7. Luc Jaulin, Isabelle Braems, Michel Kieffer, and Eric Walter. Interval methods for nonlinear identification and robust control. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, Las Vegas, Nevada, 2002.
8. Luc Jaulin, Michel Kieffer, Olivier Didrit, and Eric Walter. *Applied Interval Analysis*. Springer, 2001.
9. Lalo Magni, Riccardo Scattolini, and Karl Johan Åström. Global stabilization of the inverted pendulum using model predictive control. In *Proceedings of the 15th IFAC World Congress*, Barcelona, 2002.
10. Stefano Malan, Mario Milanese, and Michele Taragna. Robust analysis and design of control systems using interval arithmetic. *Automatica*, 33(7):1363–1372, 1997.
11. Ramon E. Moore. Methods and applications of interval analysis. *Philadelphia, SIAM*, 1979.
12. Tarek Raïssi, Nacim Ramdani, and Yves Candau. Guaranteed stated estimation for nonlinear continuous systems with taylor models. In *Proceedings of the IFAC Symposium on System Identification (SYSID)*, 2002.